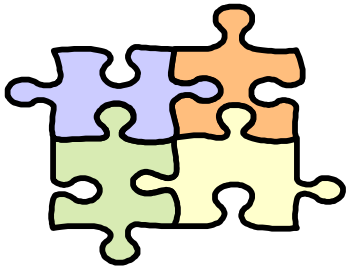


Programming the Grid: Distributed Software Components & Grid Web Services for Scientific Applications

Dennis Gannon

Department of Computer Science

And

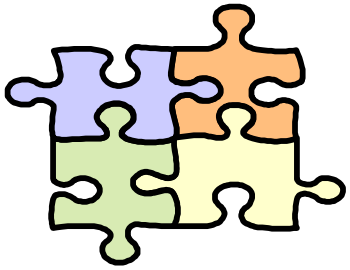
Pervasive Technology Laboratories

Indiana University

And

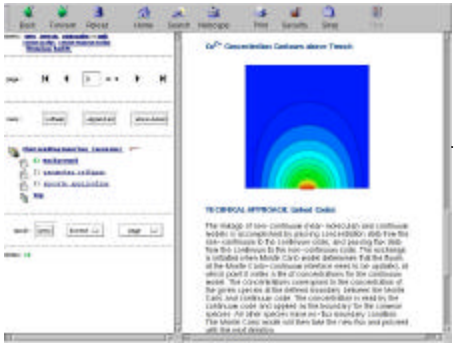
NCSA Alliance

Outline



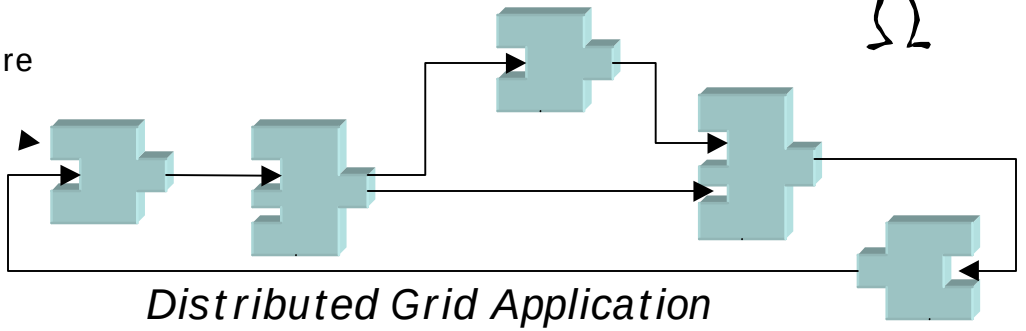
- Component Technologies
  - Concept
  - DOE CCA model
    - Grid implementation
- Web Services
  - Application Level Services
  - Components as Services
- Putting it together
  - challenges

Grid Layers



User Portals/ Science Portals

Launch, configure
And control



Distributed Grid Application

Grid Web Services Layer

P2P / Grid Services Layer

Peer Creation & resolution Services

Discovery service

Information/ Naming Services

(co-)scheduling Service

Security Service

Accounting Service

Information Routing

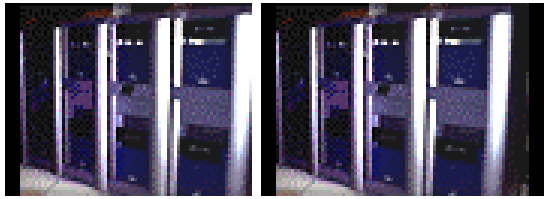
Event/Mesg Service

User Help Services

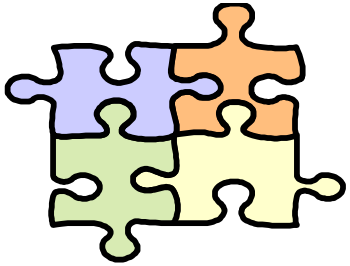
Monitoring Service

Resource layer

1000s of PCs ->massive supercomputers



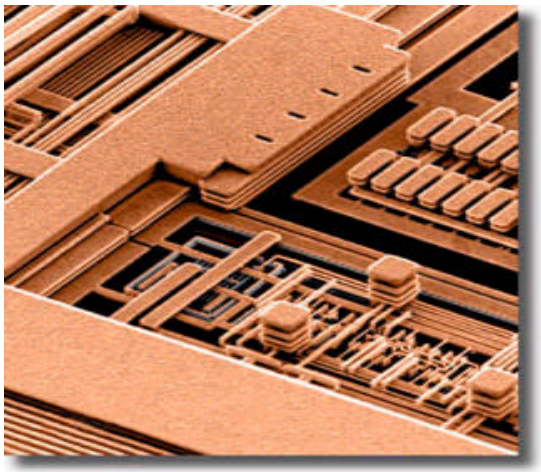
Some Examples

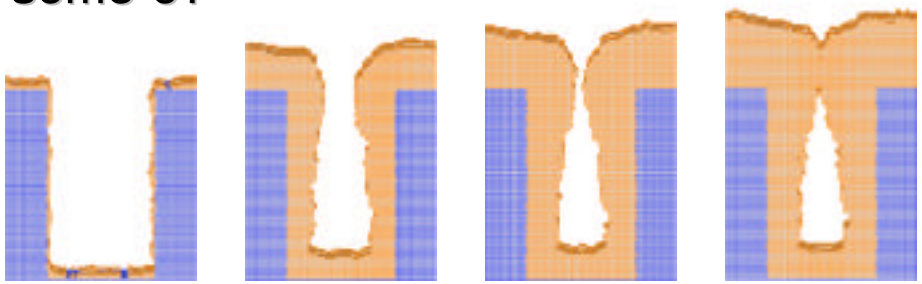


- To see what is required for a Grid programming, need to look at real examples.
  - Materials Science
  - Remote instrument control
  - Orbiting Space Junk

An Industrial Application

- Materials Science
  - NCSA Chemical Eng. Team
  - Semiconductor Processing
    - Copper metalization on Silicon
    - Optimizing Semicon Fab Processes
  - The Problem:
    - Process Engineer at company X wants to try an experiment
      - to simulate proposed changes in chip fabracation by changing some of process parameters.
      - To accomplish this two different simulations needed to be coupled.

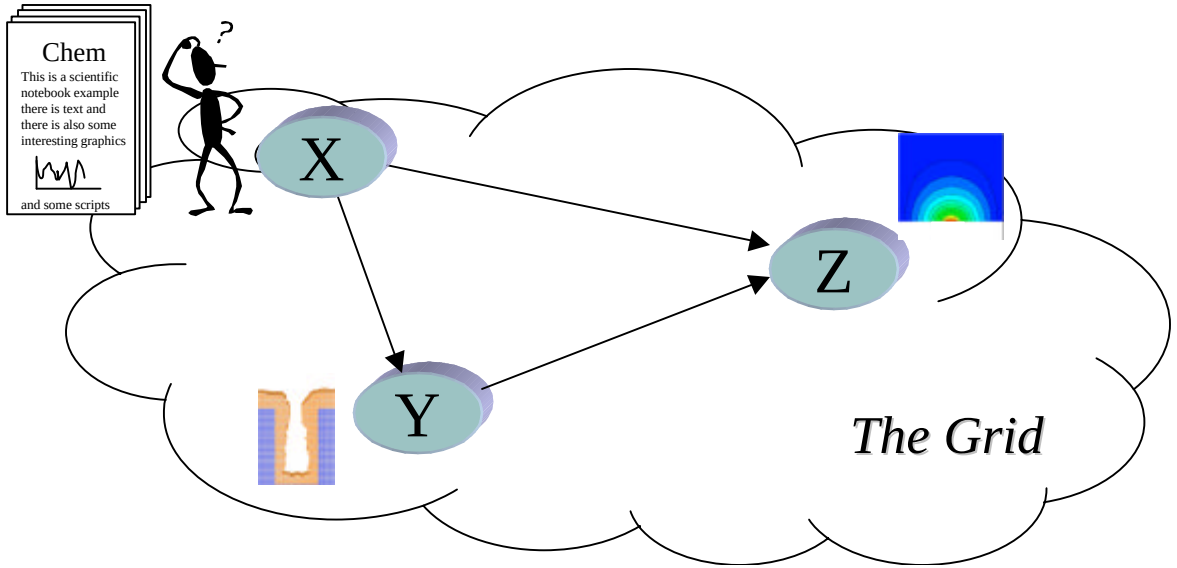




The Problem in Greater Depth

- It is a Grid coupled multi-scale simulation problem
  - The Monte Carlo simulation is at University Y and the Continuum simulation is at Company Z
- They decide to collaborate but
  - 1. The effort must protect security for each peer.
    - Codes may be used but not moved
    - Data must be protected.
  - 2. How does Engineer at X do this?

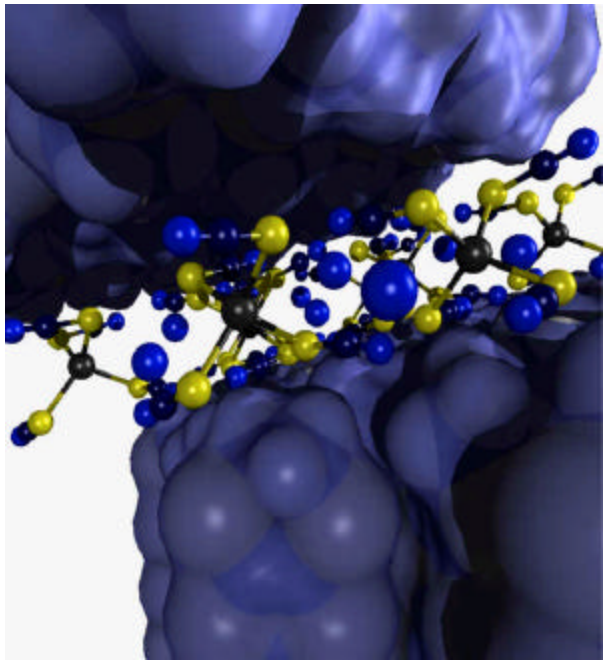
Engineer's Grid Script
1 Launch MC at Y
2. Launch Cont at Z
3. Link MC to Cont
4. Execute MC & Cont
5. Store results at X



Application 2: X-Ray Crystallography

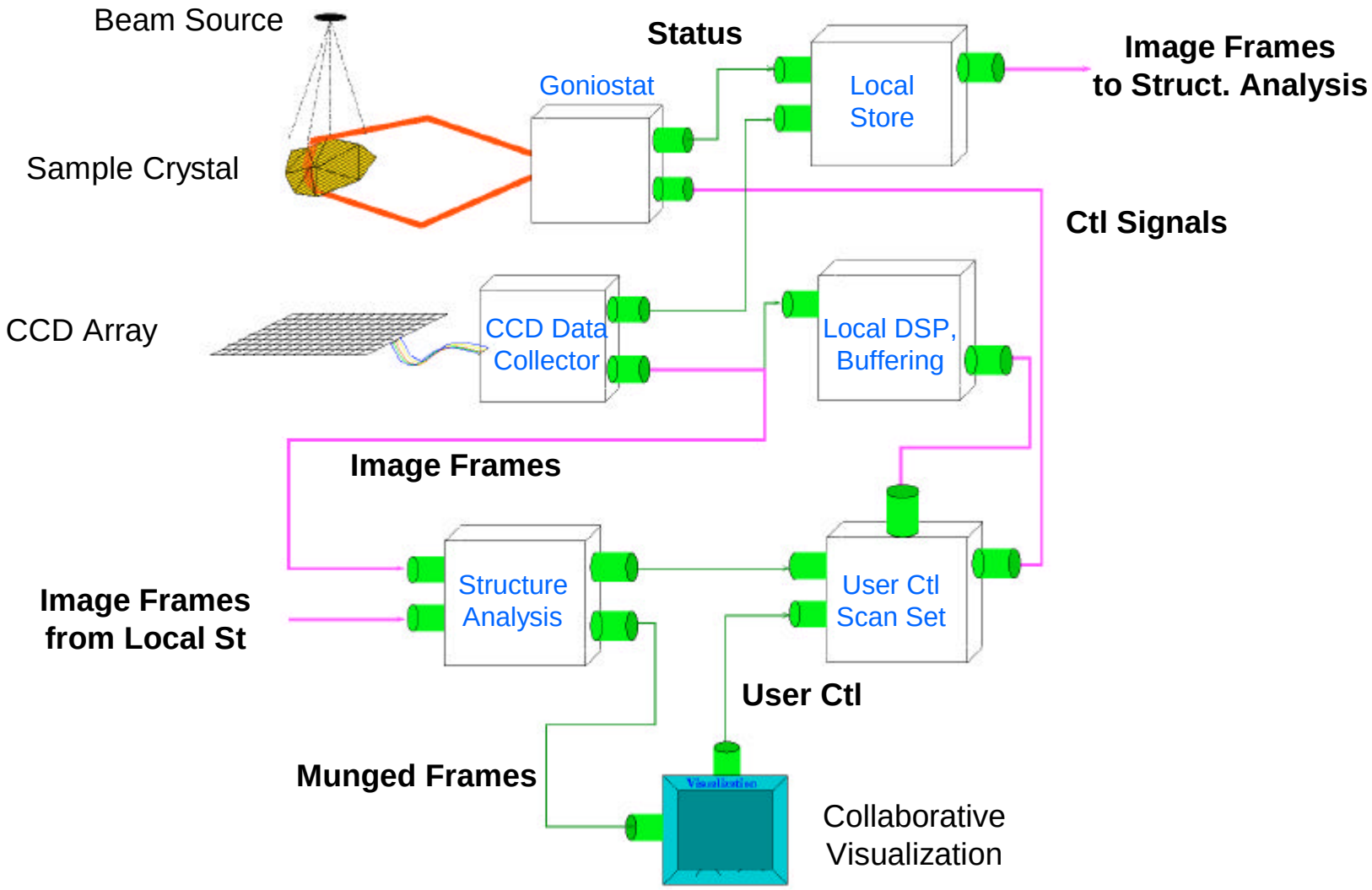
Xports Project:
Network-Based Macro-Molecular Structure Determination

Remote Instrument Control
And on-line data analysis



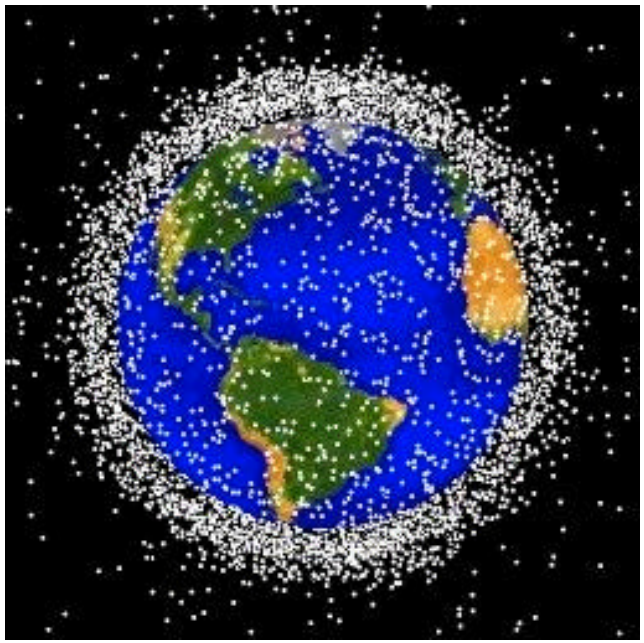
R. Bramley, D. F. McMullen, J. Huffman
Indiana University
I. Foster, G. von Laszewski, M. Westbrook
Argonne National Laboratory
E. Westbrook,
Lawrence Berkeley Laboratory

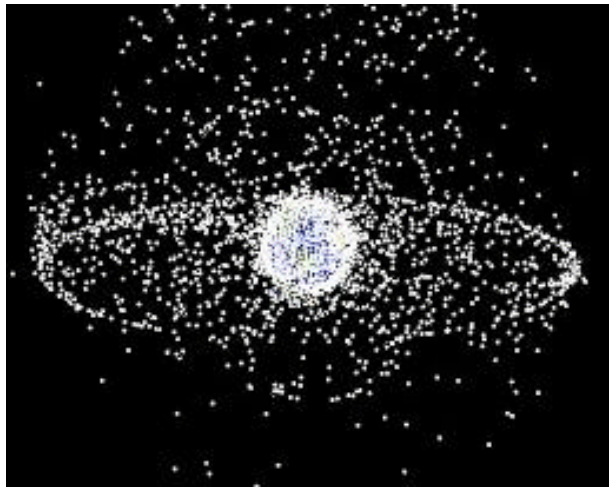
Application



Application 3: Tracking Space Junk

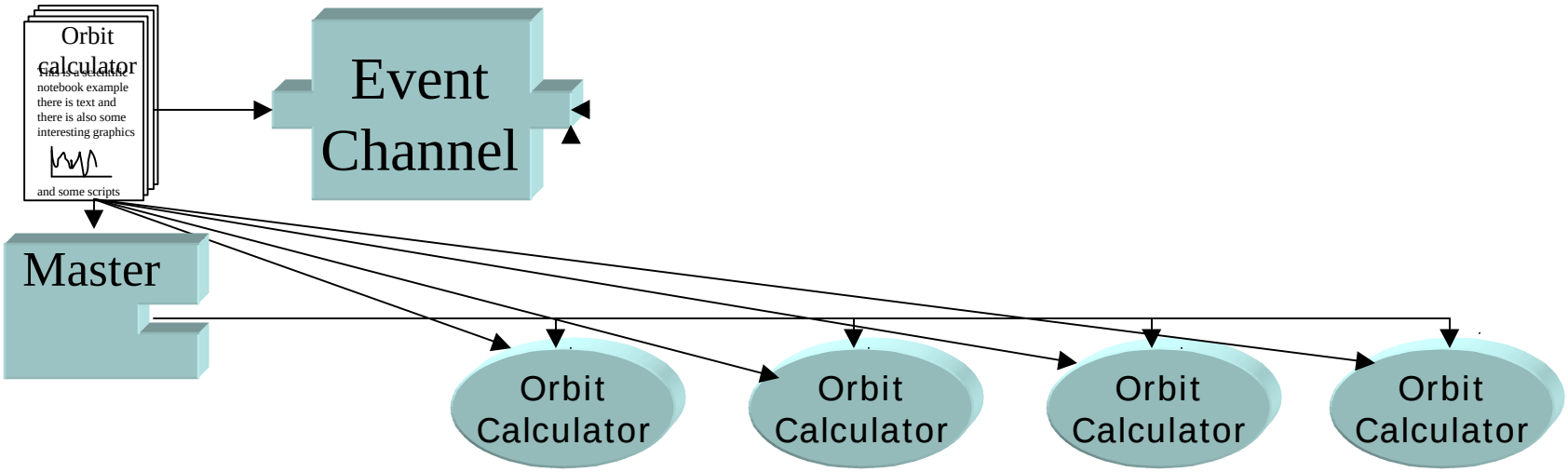
- Approximately 9,000 objects larger than 3 cm orbiting earth with speeds averaging about 10 Km/sec
- The Problem:
  - To launch something you need to know if you will hit the space junk.
  - Must predict thousands of orbit paths.
- (an homage to Entropia and Condor)



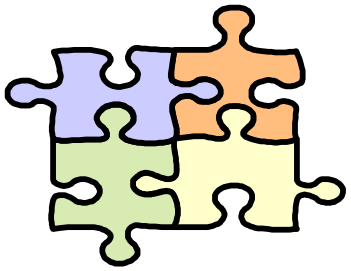


Computation Based on Peer to Peer Model

- The Algorithm
  - Divide objects into groups of size < 100
  - Send each group to an idle machine to compute orbit trajectories
  - Report potential collisions back to root component.
  - Launch master and workers
  - Load tasks into workers. Workers report results back to user as collision events.

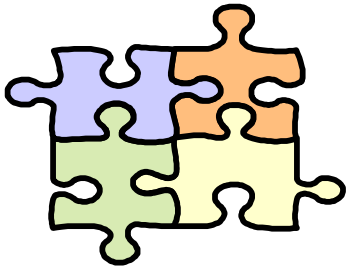


Software Components



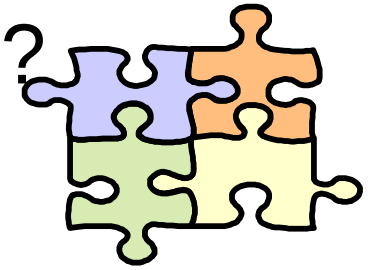
- What is a Component Architecture?
  - A Software Engineering Methodology/Standard
    - to promote code reusability & reduce application complexity.
    - a precise set of protocol that define interoperability between and behavior of software "components" of an application.
  - A Component
    - Is an encapsulated software module defined by its public interfaces.
  - A Component framework is -
    - A compile-time/runtime environment for instantiating, composing and running components.

Examples of CAs:



- Desktop Software:
  - Simplify the design and interoperability of desktop apps
    - Microsoft COM, Java Beans, Gnome CORBA Components
- Distributed Software:
  - Make it possible to easily incorporate a remote object into a local computation
    - Enterprise Java Beans, DCOM/.Net, CORBA Component Model (CCM).

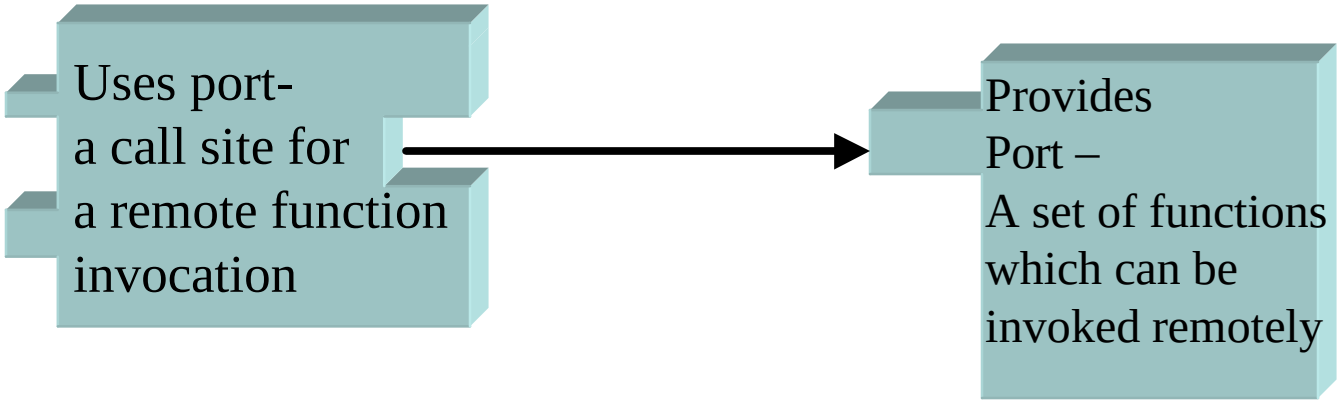
What about CAs for the Grid?



- Several good projects
  - Corba Based CAs: IRISA/INRIA, Gateway, Miss. State, Rutgers, NASA Glenn
  - Imperial College
- Look at US Dept. of Energy CCA.
  - Designed for both parallel and Dist. Apps.

The DOE Common Component Architecture

- A specification for a CA for large scale scientific computation
  - Both a DOE project and an open forum
    - Sandia, llnl, argonne, pnnl, ornl, nasa (icase), utah, indiana, Maryland, ncsa, … about 50 people.
- Key Idea: dynamic composition by linking "ports"
  - Provides ports: interface of "services" provided
  - Uses ports: a call-site/use of a service to be provided by another coponent.



Building Applications by Composition

- Connect uses Ports to Provides Ports.

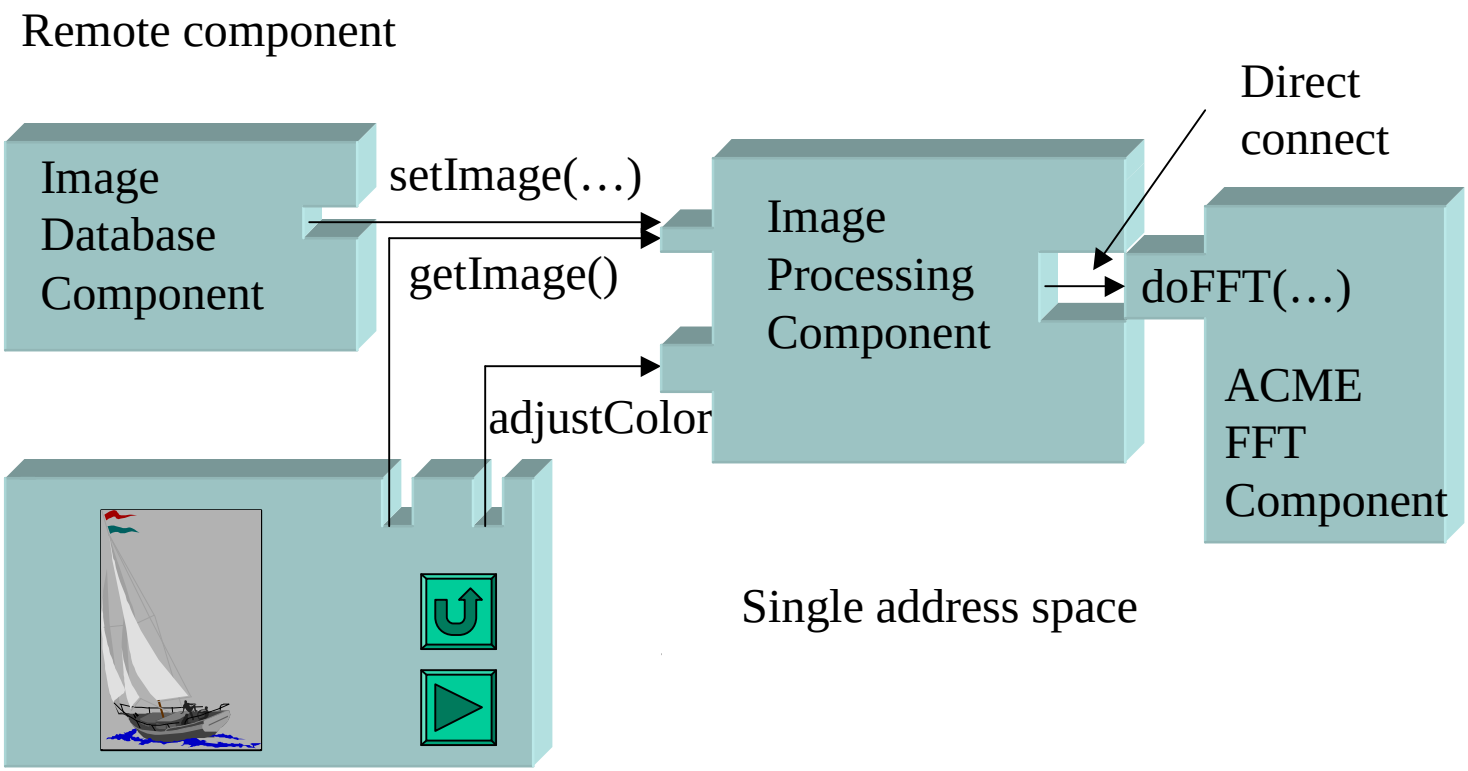
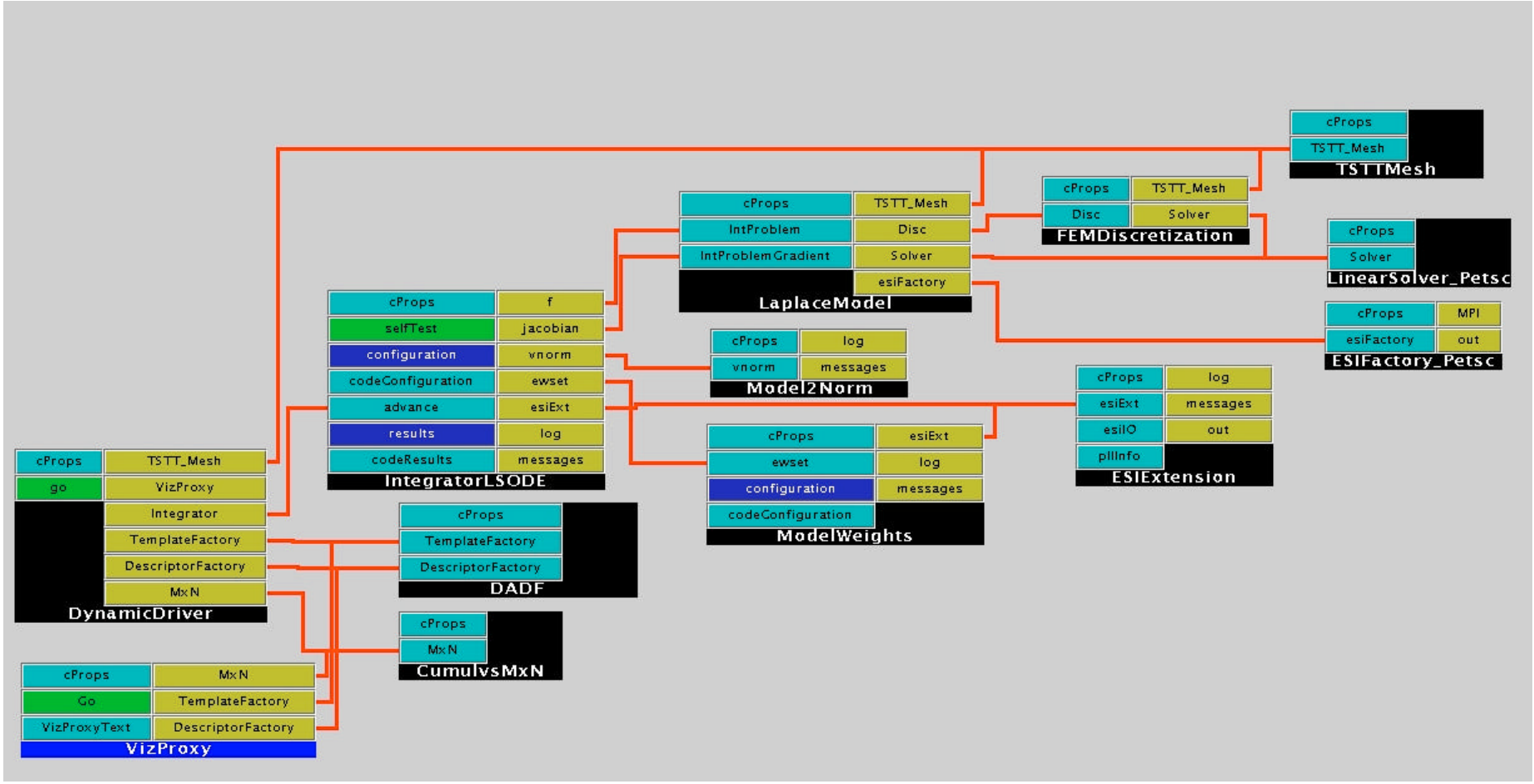


Image tool graphical interface component

CCA Application Example

Solving a Time Dependent PDE from an Unstructured FEM Mesh

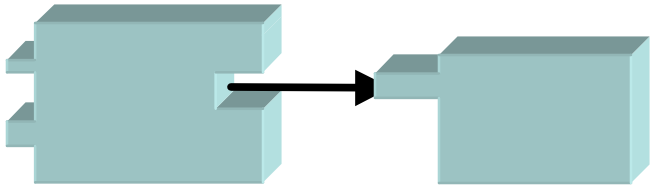
Lois McInnes, Lori Freitag, Boyana Norris, David Bernholdt, Jim Kohl
Ben Allen and Rob Armstrong



Using components on the Grid

- How do you find out about existing application components?
- How to launch a remote component?
- How to discover remote instances?
- What protocol should be used for remote method invocations?
- Can components use “messages/events” to communicate?
- How to manage a large distributed set of remote components?
- How to handle authentication and authorization?
- How does one encapsulate legacy applications?

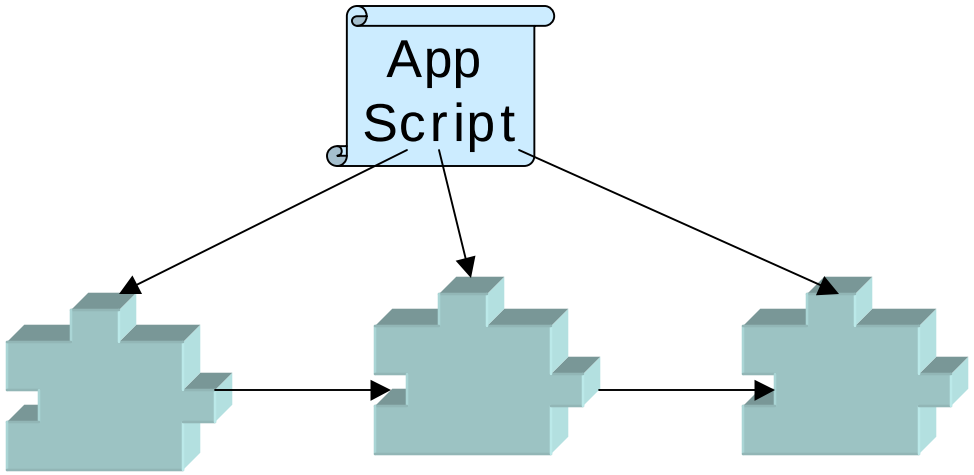
Component Communication



- How do components communicate?
  - Use a simple Remote Procedure Call Mechanism
    - We Use Simple Object Access Protocol SOAP
      - Uses XML over HTTP and XML Schemas.
      - Java RMI implemented over SOAP and C++ bindings
      - .NET compatible. Oh, Boy! Also works well with WSDL.
      - Very robust but slooooow.
  - Events/Messages
    - Objects encoded as XML Docs.
      - Compatible with Soap

How to link and control a set of components?

- We use a Jython Script
  - Script uses java cog to launch the component instances through gram
    - Or script can consult a directory service
  - The script can call a connect operation to link two instances
  - The script can invoke ports directly



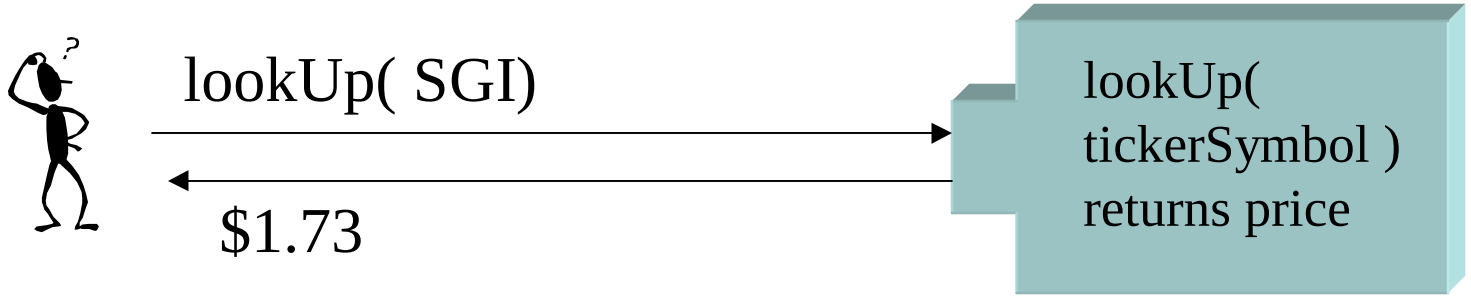
Events

- Event/Notification Services are an essential part of all distributed systems.
- Application Events
  - "I am done." "I just wrote to a file", "here is a result" "Oops! I just did an underflow"
- Event Publishers & Listeners
  - Some Listeners subscribe; others poll
  - Must have persistent event channels
    - grid web services that allow portals access to application history logs.

Security

- The Problem
  - When two components are linked how does the provider authenticate the user?
  - When I wish others to use my component instance how do I handle authorization?
    - Must be done on a port-by-port basis.
    - Same problems as faced by web services
- Solutions
  - Using SSL with globus proxy certs.
  - Components have a “control port” which can be used to load authorized user lists.
    - Legion/Akenti/GAA projects have some good lessons to teach us.

Something New: *Web Services*



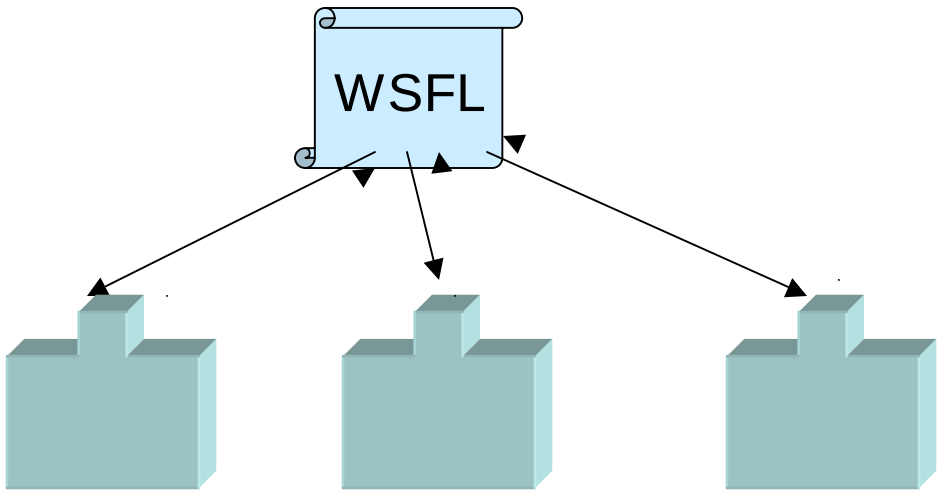
- A Web Service is
  - A network service that provides a functional interface to remote clients.
    - designed to allow composition of services
- Big collaboration between big players
  - IBM, Microsoft, Oracle, HewlitPaq, … , sun
    - Central to MS .NET and IBM services plans
- In effect, a web service is an instance of a CCA Grid component with only “provides ports”.

Central Web Services Standards

- Web Services Description Language (WSDL)
  - An XML document that
    - Describes the interfaces types for each port
      - The contents of messages it receives and sends
    - Describes the bindings of interfaces to protocols
      - SOAP – XML over HTTP is default, others possible
      - Describes the access points (host/port) for the protocol bindings
- Web Services Flow Langauge (WSFL)
  - An XML document that describes how a business process invokes a set of services
- UDDI & WSIL
  - A mechanism to discover WSDL docs for services
- WSIF –
  - Translates WSDL docs into Java proxies

WSFL & Control Flow

- An XML document that describes the work flow of a task that uses web services
  - Completely analogous to component script
  - Except no direct service to service link.

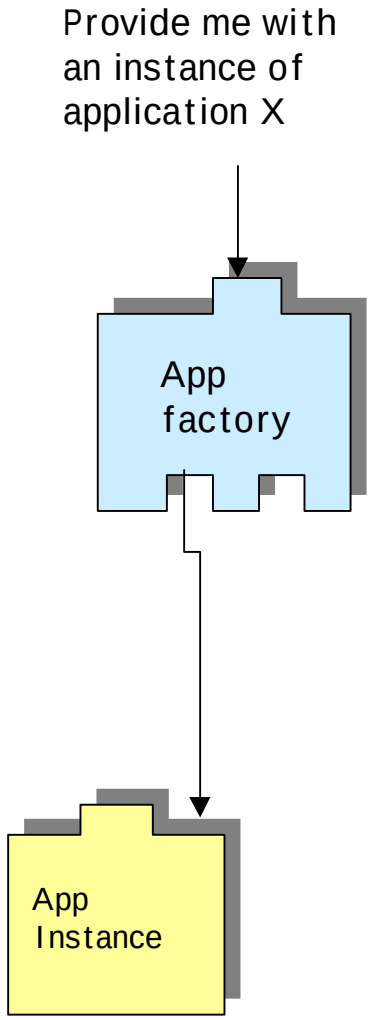


Web Services in Grid Programming

- Authentication and Authorization
  - Obvious examples of Grid Web Services
- Discovery Service
  - Finding remote “instances” of things
    - May be LDAP based like MDS or other …
- Resource Brokering
  - What are the best resource upon which to run my application now?
- Application Factory Service

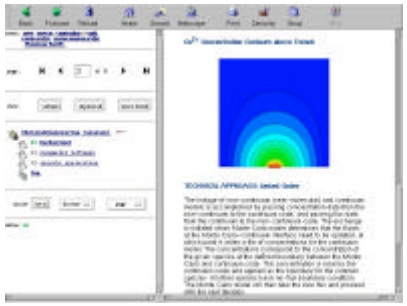
Application Factory Service

- A service that understands a how to instantiate a running instance of an app component.
  - You provide it with appropriate requirements initial conditions, etc.
  - The service
    - checks you credentials and authorization
    - launches the app and returns a reference to it.



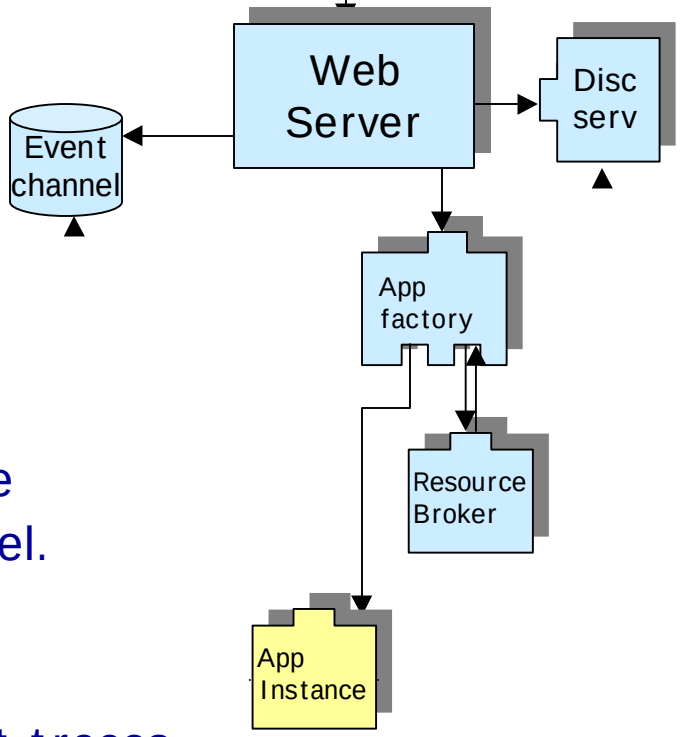
Anatomy of a Componentized Grid Service Application

- User Configures Application from Web Browser
  - Sets application parameters
  - Launches job



User Portals/ Science Portals

- Web Server contacts appropriate application factory service (FS)
  - Supplies FS with task parameters
  - FS contacts Resource Broker and secures job launches.
  - Returns jobID to server to browser

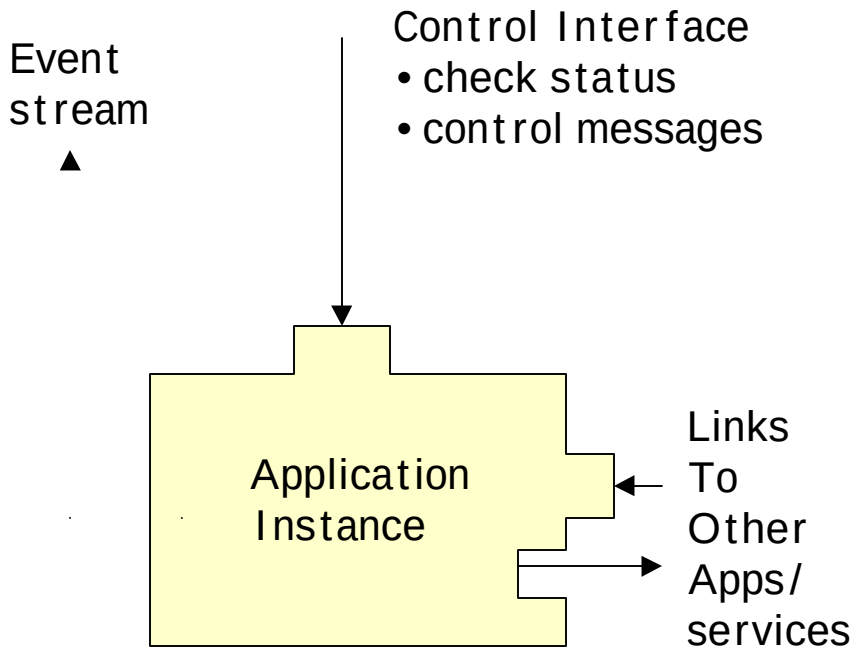


- Job begins execution
  - Publishes its contact point to discovery service
  - Begins publishing status events to event channel.

- Web server discovers application
  - Allows user to interrogate it or retrieve event traces

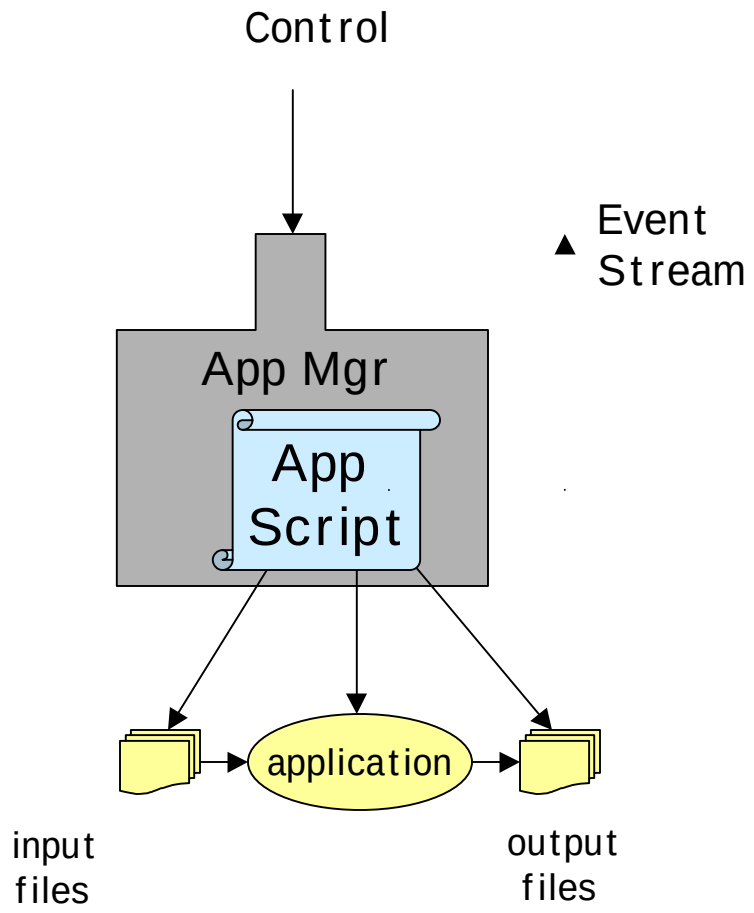
An Application Instance

- Typical Instance
  - Publishes event stream to “well known” channel.
  - Has a Control Interface to allow portal level interrogation or remote steering
  - May be linked to other components/services



Encapsulating Legacy Apps

- Common Case
  - Legacy App that reads files and writes files.
  - Use a “scripted component” called an app manager.
    - Component runs a python script loaded at startup or through a control command
  - The App Script
    - Stages files
    - Launches and monitors application
    - Writes Output files
    - publishes event streams



Conclusions

- Component models can be an effective way to design grid applications
  - Easy integration with globus grids
  - An event/messaging model is critical
  - Not hard to encapsulate legacy applications
  - Components can be composed and managed by simple grid scripts
- Completely consistent with Web Services model for grid application.
  - CCA grid component IS a transient web service
  - Application factory services a useful tool for building distributed applications and managing portals.